Sapthagiri College of Engineering

Dept. of Information Science and Engineering

Circular message:

Technical Coding Contest

In this pandemic we bring, only for you, an online coding event to build-up your coding skills and unleash your talent. We at I.S. technical club are determined to provide you with more opportunities in the world of coding, hence we are here with some interesting programming questions for your creative mind. Let's quick start your imagination. With the basic yet interesting questions where anybody can code and learn.

Click on the link to participate:

Second Year: “https://www.hackerrank.com/2ndyearcontest”
Third Year: “https://www.hackerrank.com/technical-coding-contest-third-year”

This link can be used to access the test after the registration too.
DATE: 25th Oct, 2020
TIMING: 11 AM to 1 PM.
E-Certificate will be given to the winners only

I.S. TECHNICAL CLUB
INFORMATION SCIENCE TECHNICAL CLUB
I'M THE BEST CODER!
Technical Coding Contest
OCTOBER 25th, 11 AM TO 1 PM
PLATFORM - HackerRank
Prizes - Certificate to Winners
"Make it work,make it right,make it fast."

Questions and answers of contest (Second Year):

question 1

In this challenge, you are required to calculate and print the sum of the elements iarray, keeping in mind that some of those integers may be quite large.

Function Description

Complete the *aVeryBigSum* function in the editor below. It must return the sum of all array elements.

aVeryBigSum has the following parameter(s):

- *int ar[n]*: an array of integers .

Return

- *long*: the sum of all array elements

Input Format

The first line of the input consists of an integer .
The next line contains space-separated integers contained in the array.

Output Format

Return the integer sum of the elements in the array.

Answer)
```
import math

import os

import random

import re

import sys

# Complete the aVeryBigSum function below.

def aVeryBigSum(ar):

  sum = 0

  for i in ar:

  sum += i
```

```
    return sum
if __name__ == '__main__':
    fptr = open(os.environ['OUTPUT_PATH'], 'w')
  ar_count = int(input())
ar = list(map(int, input().rstrip().split()))
result = aVeryBigSum(ar)
fptr.write(str(result) + '\n')
fptr.close(
```

<u>question 2)</u>Given a square matrix, calculate the absolute difference between the sums of its diagonals.

For example, the square matrix arr is shown below:

```
1 2 3
4 5 6
9 8 9
```

The left-to-right diagonal = =1+5+9=15=. The right to left diagonal = 3+5+9=17=. Their absolute difference isl 15-17l=2 .

Function description

Complete the diagonalDifference function in the editor below.

diagonalDifference takes the following parameter:

- *int arr[n][m]*: an array of integers

Return

- *int*: the absolute diagonal difference

<u>answer)</u>

```
_#include<iostream>
#include<math.h>
using namespace std;
int main(){
   int n;
```

```
  cin>>n;

  int arr[n][n],larr=0,rarr=0;

  for(int i=0;i<n;++i){

  for(int j=0;j<n;++j){

    cin>>arr[i][j];

  } }

  for(int i=0;i<n;++i){

  for(int j=0;j<n;++j){

    if (i == j)

        larr += arr[i][j];

    if (i == n - j - 1)

        rarr += arr[i][j];

  }}

cout<<abs(larr-rarr);

}
```

<u>question 3)</u>You are choreographing a circus show with various animals. For one act, you are given two kangaroos on a number line ready to jump in the positive direction (i.e, toward positive infinity).

- The first kangaroo starts at location x1 and moves at a rate of v1 meters per jump.
- The second kangaroo starts at location x2 and moves at a rate of v2 meters per jump.

You have to figure out a way to get both kangaroos at the same location at the same time as part of the show. If it is possible, return YES, otherwise return NO.
For example, kangaroo 1 starts at x1=2 with a jump distance V1=1and kangaroo 2 starts at x2=1with a jump distance of v2=2. After one jump, they are both at x=3, (x1+v1=2+1,x2+v2=1+2), so our answer is YES.
Function Description

Complete the function *kangaroo* in the editor below. It should return YES if they reach the same position at the same time, or NO if they don't.

kangaroo has the following parameter(s):

- *x1, v1*: integers, starting position and jump distance for kangaroo 1
- *x2, v2*: integers, starting position and jump distance for kangaroo 2

Input Format

A single line of four space-separated integers denoting the respective values of x1,v1 ,x2 , and v2.

answer)

```
using namespace std;

vector<string> split_string(string);

// Complete the kangaroo function below.

string kangaroo(int x1, int v1, int x2, int v2) {

if (v1 > v2) {

int remainder = (x1 - x2) % (v2 – v1);

if (remainder == 0) {

return "YES";

}

}

    return "NO";

}

int main()

{

  ofstream fout(getenv("OUTPUT_PATH"));

 string x1V1X2V2_temp;

 getline(cin, x1V1X2V2_temp);
```

```
vector<string> x1V1X2V2 = split_string(x1V1X2V2_temp);

 int x1 = stoi(x1V1X2V2[0]);

 int v1 = stoi(x1V1X2V2[1]);

 int x2 = stoi(x1V1X2V2[2]);

 int v2 = stoi(x1V1X2V2[3]);

string result = kangaroo(x1, v1, x2, v2);

fout << result << "\n";

 fout.close();

return 0;

}

vector<string> split_string(string input_string) {

    string::iterator new_end = unique(input_string.begin(), input_string.end(), [] (const char &x,
const char &y) {

        return x == y and x == ' ';

    });

 input_string.erase(new_end, input_string.end());

 while (input_string[input_string.length() - 1] == ' ') {

        input_string.pop_back();

    }

  vector<string> splits;

    char delimiter = ' ';

  size_t i = 0;

    size_t pos = input_string.find(delimiter);

    while (pos != string::npos) {

        splits.push_back(input_string.substr(i, pos – i));
 i = pos + 1;

      pos = input_string.find(delimiter, i);

  }  splits.push_back(input_string.substr(i, min(pos, input_string.length()) - i + 1));

return splits;

}
```

question 4)

Lexicographical order is often known as alphabetical order when dealing with strings. A string is *greater* than another string if it comes later in a lexicographically sorted list.

Given a word, create a new word by swapping some or all of its characters. This new word must meet two criteria:

- It must be greater than the original word
- It must be the smallest word that meets the first condition

For example, given the word w=abcd, the next largest word is abcd.

Complete the function *biggerIsGreater* below to create and return the new string meeting the criteria. If it is not possible, return no answer.

Function Description

Complete the *biggerIsGreater* function in the editor below. It should return the smallest lexicographically higher string possible from the given string or no answer.

biggerIsGreater has the following parameter(s):

- *w*: a string

Input Format

The first line of input contains T , the number of test cases.
Each of the next T lines contains w.

answer)

```
using namespace std;

// Complete the biggerIsGreater function below.

string biggerIsGreater(string s) {
```

```
bool result=next_permutation(s.begin(), s.end());

if(result==true){

    return s;

}

    else{

        return "no answer";

    }

}

int main()

{

    ofstream fout(getenv("OUTPUT_PATH"));

     int T;

    cin >> T;

    cin.ignore(numeric_limits<streamsize>::max(), '\n')

 for (int T_itr = 0; T_itr < T; T_itr++) {

        string w;

        getline(cin, w);

string result = biggerIsGreater(w);

fout << result << "\n";

    } fout.close();

return 0;

}
```

Questions and answers of contest (Third Year):

question 1

In this challenge, you are required to calculate and print the sum of the elements iarray, keeping in mind that some of those integers may be quite large.

Function Description

Complete the *aVeryBigSum* function in the editor below. It must return the sum of all array elements.

aVeryBigSum has the following parameter(s):

- *int ar[n]*: an array of integers .

Return

- *long*: the sum of all array elements

Input Format

The first line of the input consists of an integer .
The next line contains space-separated integers contained in the array.

Output Format

Return the integer sum of the elements in the array.

Answer)
```
import math

import os

import random

import re

import sys

#Complete the aVeryBigSum function below.

def aVeryBigSum(ar):

  sum = 0

  for i in ar:

  sum += i
  return sum
if __name__ == '__main__':
  fptr = open(os.environ['OUTPUT_PATH'], 'w')
 ar_count = int(input())
ar = list(map(int, input().rstrip().split()))
result = aVeryBigSum(ar)
fptr.write(str(result) + '\n')
fptr.close(
```

question 2) Given a chocolate bar, two children, Lily and Ron, are determining how to share it. Each of the squares has an integer on it.

Lily decides to share a contiguous segment of the bar selected such that:

- The length of the segment matches Ron's birth month, and,
- The sum of the integers on the squares is equal to his birth day.

You must determine how many ways she can divide the chocolate.

Consider the chocolate bar as an array of squares, s=[2,2,1,3,2]. She wants to find segments summing to Ron's birth day, d=4with a length equalling his birth month, m=2. In this case, there are two segments meeting her criteria: [2,2] and [1,3].

Function Description
Complete the *birthday* function in the editor below. It should return an integer denoting the number of ways Lily can divide the chocolate bar.

birthday has the following parameter(s):

- *s*: an array of integers, the numbers on each of the squares of chocolate
- *d*: an integer, Ron's birth day
- *m*: an integer, Ron's birth month

Input Format

The first line contains an integer , the number of squares in the chocolate bar.
The second line contains space-separated integers , the numbers on the chocolate squares where 0<i<n.

The third line contains two space-separated integers, d and m, Ron's birth day and his birth month

answer)

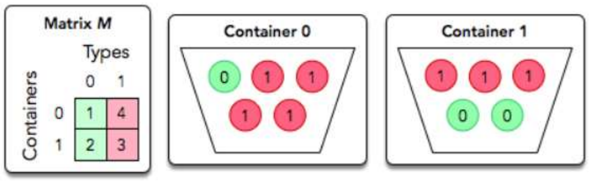
```
import math

import os

import random

import re

import sys

#Complete the birthday function below.

def birthday(s, d, m):

    count=0

    length = len(s)

    for i in range(0,length-m+1):

        if sum(s[i:i+m]) == d:

            count +=1

    return count

if __name__ == '__main__':

    fptr = open(os.environ['OUTPUT_PATH'], 'w')

  n = int(input().strip())

  s = list(map(int, input().rstrip().split()))

dm = input().rstrip().split()

d = int(dm[0])

m = int(dm[1])

result = birthday(s, d, m)

fptr.write(str(result) + '\n')
```

fptr.close()

question 3)

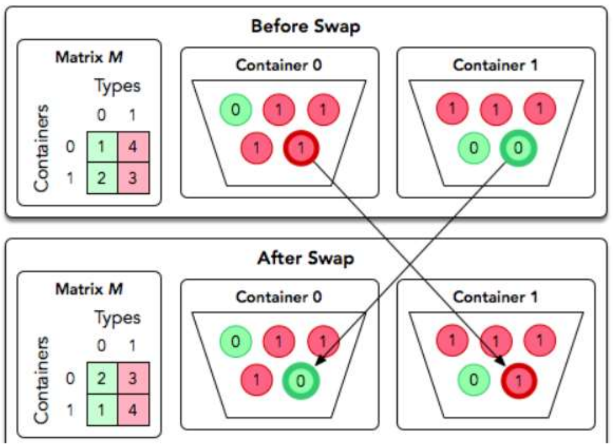
David has several containers, each with a number of balls in it. He has just enough containers to sort each type of ball he has into its own container. David wants to sort the balls using his sort method.

As an example, David has $n = 2$ containers and 2 different types of balls, both of which are numbered from 0 to $n - 1 = 1$. The distribution of ball types per container are described by an $n \times n$ matrix of integers, $M[container][type]$. For example, consider the following diagram for $M = [[1, 4], [2, 3]]$:



In a single operation, David can *swap* two balls located in different containers.

The diagram below depicts a single swap operation:



David wants to perform some number of swap operations such that:

- Each container contains only balls of the same type.
- No two balls of the same type are located in different containers.

You must perform q queries where each query is in the form of a matrix,M . For each query, print Possible on a new line if David can satisfy the conditions above for the given matrix. Otherwise, print Impossible.

Function Description

Complete the *organizingContainers* function in the editor below. It should return a string, either Possible or Impossible.

organizingContainers has the following parameter(s):

- *containter*: a two dimensional array of integers that represent the number of balls of each color in each container

Input Format

The first line contains an integer , the number of queries.

Each of the next sets of lines is as follows:

1. The first line contains an integer , the number of containers (rows) and ball types (columns).
2. Each of the next lines contains space-separated integers describing row .

answer) vector<bool> isprime(N+ 1, true);

```
void prime()
{
  rep(i, 2, sqrt(N))
  {
    if (isprime[i])
    {
      for (ll j = i * i; j <= N; j += i)
      {
        isprime[j] = false;
      }
    }
  }
  return;
```

```
}
void solve()
{
 ll n; cin>>n;
 ll m[n][n];
 rep(i,0,n){
   rep(j,0,n){
    cin>>m[i][j];
   }
 }
 map<ll,ll>map;
 rep(i,0,n){
   ll sum=0;
   rep(j,0,n){
    sum+=m[i][j];
   }
   map[sum]++;
 }
 rep(i,0,n){
   ll sum=0;
   rep(j,0,n){
     sum+=m[j][i];
   }
   map[sum]--;
 }
 bool flag=true;
 for(auto i:map){
   if(i.second<0||i.second>0) flag=false;
 }
 if(flag) cout<<"Possible"<<endl;
 else cout<<"Impossible"<<endl;
}
int main()
{
  ios_base::sync_with_stdio(false);
  cin.tie(NULL);
  prime();
  ll t;
  cin >> t;
  while (t--)
    solve();
  return 0;
}
```

question 4) Given an array of integers, you must answer a number of queries. Each query consists of a single integer,x , and is performed as follows:

1. Add x to each element of the array, permanently modifying it for any future queries.
2. Find the absolute value of each element in the array and print the sum of the absolute values on a new line.

Tip: The Input/Output for this challenge is *very large*, so you'll have to be creative in your approach to pass all test cases.

Function Description

Complete the *playingWithNumbers* function in the editor below. It should return an array of integers that represent the responses to each query.

playingWithNumbers has the following parameter(s):

- *arr*: an array of integers
- *queries*: an array of integers

Input Format

The first line contains an integer n the number of elements in arr.
The second line contains n space-separated integers arr[i].
The third line contains an integer q, the number of queries.

The fourth line contains q space-separated integers x where queries[j]=x.

Answer)

```
vector<bool> isprime(N+ 1, true);

void prime()

{

  rep(i, 2, sqrt(N))

  {

    if (isprime[i])

    {

      for (ll j = i * i; j <= N; j += i)

      {

        isprime[j] = false;

      }

    }

  }

  return;

}
```

```
void solve()
{
      ll  n;
     cin>>n;
     vec v(n);
    rep(i,0, n) cin>>v[i];
    sort(all(v));
    vec sum(N+1, 0);
    rep(i,0, n)
        sum[i+1] = sum[i] + v[i];
    ll q;
    cin>>q;
    ll add=0;
    rep(i,0, q) {
        ll x; cin>>x;
        add += x;
        ll k = lower_bound(all(v), -add) - v.begin();
        ll ans = 0;
        ans += (sum[n] - sum[k]) + add * (n - k);
        ans += -((sum[k] - sum[0]) + add * k);
        cout<<ans<<endl;
    }
}
```

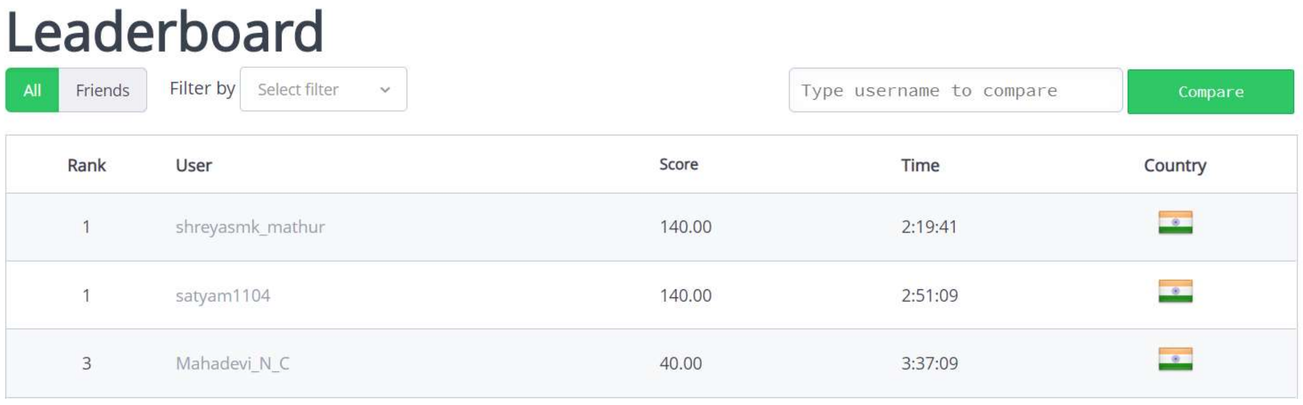
```
int main()
{
  ios_base::sync_with_stdio(false);
  cin.tie(NULL);
  // prime();
  // ll t;
  // cin >> t;
  // while (t--)
    solve();
  return 0;
}
```

WINNERS :

1. **Avinash kumar(Second Year)**
2. **Shreyas M kaushik(Third Year)**

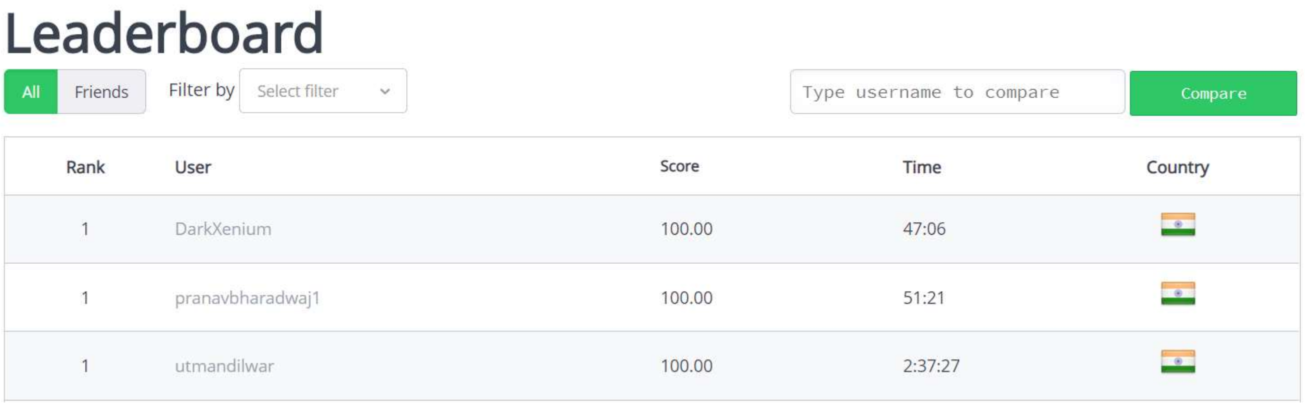
Pictures :

Third Year Top 3 Participants



Rank	User	Score	Time	Country
1	shreyasmk_mathur	140.00	2:19:41	
1	satyam1104	140.00	2:51:09	
3	Mahadevi_N_C	40.00	3:37:09	

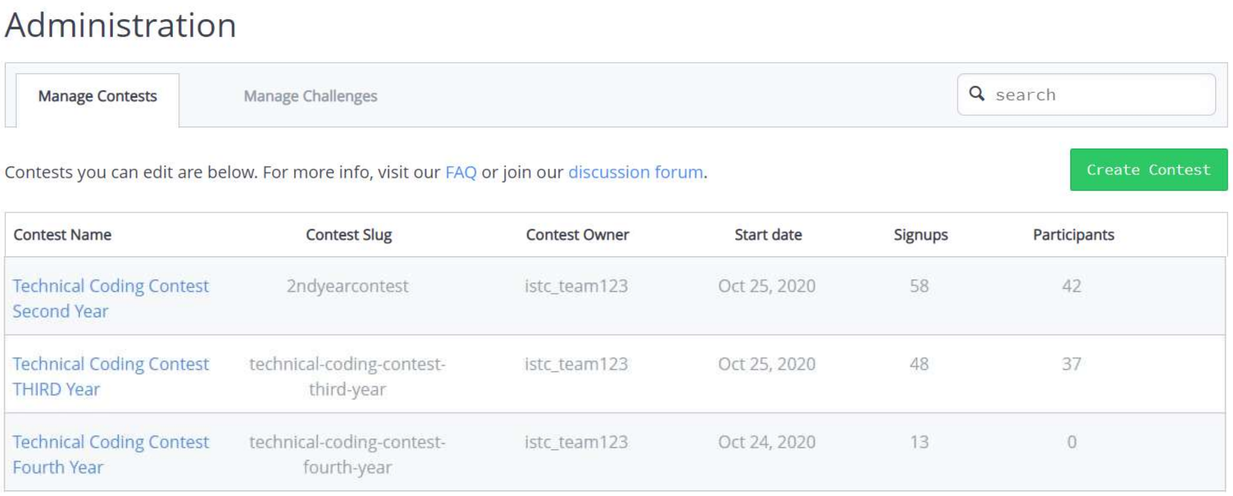
Second Year Top 3 Participants



Leaderboard

Rank	User	Score	Time	Country
1	DarkXenium	100.00	47:06	
1	pranavbharadwaj1	100.00	51:21	
1	utmandilwar	100.00	2:37:27	

CONTEST PORTAL



Administration

Contests you can edit are below. For more info, visit our FAQ or join our discussion forum.

Contest Name	Contest Slug	Contest Owner	Start date	Signups	Participants
Technical Coding Contest Second Year	2ndyearcontest	istc_team123	Oct 25, 2020	58	42
Technical Coding Contest THIRD Year	technical-coding-contest-third-year	istc_team123	Oct 25, 2020	48	37
Technical Coding Contest Fourth Year	technical-coding-contest-fourth-year	istc_team123	Oct 24, 2020	13	0

Sapthagiri College of Engineering

Dept. of Information Science and Engineering

TECHNICAL CLUB REPORT

The **TECHNICAL CODING CONTEST** activity was conducted by the I.S. technical club and was organized on 25/10/2020.The activity was distributed into 2 groups.

- Technical Contest for second year
- Technical Contest for third year

The contest was conducted online through the website "www.hackerrank.com" on the day from 11 am to 1 pm. The total number of students attended the activities were 100 plus. We saw much of enthusiasm among students and many of them asked for more of such activities.

The winners of each group are as follows:

1. Avinash kumar(Second Year)
2. Shreyas M kaushik(Third Year)

We would like to thank our HOD sir for letting us organizing this event in our department; we would like to thank our event faculty coordinator Sudarsanan D sir and our Management to provide us the permission and help to organize this event in our college.

Students Reviews:

Shruti Pandey (Second year): This event was really knowledgeable which helped me to improve and enhance my coding skills. The support from coordinator was good they communicated to us regarding this(informed and reminded us of this event).Question chosen were also good and were good for revising the learn concepts.
Would like to thank ISTC team for conducting such a good event for us.

Utkarsh Mandilwar (Second year): I would like to thank the I.S. technical club for conducting the coding competition among students. It was a nice experience for us, coding on an online platform has opened new challenges and opportunities for us, at the same time I hope you will conduct much more competitions and tech interactions like this which will help us get an edge in this competitive world.

Outcomes:

The students will now be well prepared by the time they take up placement activities in final year being experienced about the coding contest the level of competition they may face to get a job. Also they can prepare and try different strategies to attempt contests like these. Their basics tend to be stronger than before as they are practically applying the concepts rather than just reading, memorizing algorithms and not really know what they could do with them in real world scenarios

Sudarsanan D

(Faculty Coordinator)

Event coordinators:

1. **Vivek Singh Kushwaha (1SG18IS124)**
2. **Prateek Kumar Singh (1SG18IS080)**
3. **Sanjana Sinha (1SG19IS090)**
4. **Shreya Ranjan (1SG19IS096)**

H.O.D

(Dr. H.R. Ranganatha)

Dept. of Information Science & Engg.